

CLAIMS

1. A countermeasure method for implementation in an electronic component and implementing a public-key cryptography algorithm comprising exponentiation computation of the type $y=g^d$, where g and y are elements of the determined group G written in multiplicative notation, and d is a predetermined number, said countermeasure method being characterized in that it comprises a masking first step for expressing the exponent d randomly in the form $d=d_2.s+d_1$, where d_1 , d_2 , and s are integers and a second step for computing the value of $y=g^d$ in G by any double exponentiation algorithm of the type $(g^{d_1}).(h^{d_2})$ with $h=g^s$ in G .

2. A countermeasure method according to claim 1, characterized in that the group G is written in additive notation.

3. A countermeasure method according to claim 1, characterized in that the method comprises the following steps:

1) Masking of d :

1a) Express d randomly in the form $d=d_2.s+d_1$, where d_1 , d_2 , and s are integers

1b) Let $(d_1(t), d_1(t-1), \dots, d_1(0))$ and $(d_2(t), d_2(t-1), \dots, d_2(0))$ be the respective binary representations of d_1 and of d_2

2) Double exponentiation:

2a) Define (compute) the element $h=g^s$ in G

2b) Initialize the register A with the neutral element of G

2c) For i from t down to 0, do the following:

2c1) Replace A with A^2

5 2c2) If $d_1(i)=1$, replace A with $A.g$

2c3) If $d_2(i)=1$, replace A with $A.h$

2c4) Return A.

4. A countermeasure method according to claim 1, characterized in that the method comprises the following steps:

1) Masking of d:

1a) Express d randomly in the form $d=d_2.s+d_1$, where d_1 , d_2 , and s are integers

15 1b) Let $(d_1(t), d_1(t-1), \dots, d_1(0))$ and $(d_2(t), d_2(t-1), \dots, d_2(0))$ be the respective binary representations of d_1 and of d_2

2) Double exponentiation:

2a) Define (compute) the element $h=g^s$ in G

2b) Precompute $u=g.h$ in G

20 2c) Initialize the register A with the neutral element of G

2d) For i from t down to 0, do the following:

2d1) Replace A with A^2

25 2d2) If $d_1(i)=1$ and $d_2(i)=0$, replace A with $A.g$

2d3) If $d_1(i)=0$ and $d_2(i)=1$, replace A with $A.h$

2d4) If $d_1(i)=1$ and $d_2(i)=1$, replace A with $A.u$

30 2d5) Return A.

5. A countermeasure method according to claim 2, characterized in that the method comprises the following steps:

1) Masking of d:

- 5 1a) Express d randomly in the form $d = d_2 \cdot s + d_1$,
where d_1 , d_2 , and s are integers
1b) Let $(d_1(t), d_1(t-1), \dots, d_1(0))$ and
 $(d_2(t), d_2(t-1), \dots, d_2(0))$ be the respective
binary signed-digit representations for
10 d_1 and for d_2

2) Exponentiation:

- 2a) Define (compute) the point $R = s \cdot P$ in G
2b) Initialize a register A with the neutral
element of G
15 2c) For i from t down to 0, do the following:
2c1) Replace A with $2 \cdot A$
2c2) If $d_1(i)$ is non-zero, replace A
with $A + d_1(i) \cdot P$
2c3) If $d_2(i)$ is non-zero, replace A
20 with $A + d_2(i) \cdot R$
2c4) Return A.

6. A countermeasure method according to any preceding claim, characterized in that, in the masking first step, expressing the exponent d randomly in the
25 form $d = d_2 \cdot s + d_1$, where d_1 , d_2 , and s are integers, consists in choosing a random integer s and in taking d_2 equal to the default value of the integer division of d by s, and d_1 equal to the remainder of said division.

7. A countermeasure method according to any one
30 of claims 1 to 5, characterized in that expressing the

exponent d randomly in the form $d=d_2.s+d_1$, where d_1 , d_2 , and s are integers, consists in choosing a random integer d_1 , in setting s to the value 1, and in taking d_2 equal to the difference between d and d_1 .

- 5 8. An electronic component implementing the method according to any preceding claim.